

## **REMARKS**

Claims 1-5 and 24-31 are pending in the present application. Applicants respectfully request reconsideration of the claims in view of the following remarks.

### **I. Interview Summary**

Applicants' representative conducted a telephone interview with the Examiner on July 21, 2009. Applicants' representative argued that the restriction requirement does not show that the inventions are distinct and that the Examiner has not established a serious burden. No agreement was reached. Applicants' representative conducted a telephone interview with the Examiner's Supervisory Patent Examiner, Pierre Vital, on July 24, 2009. Mr. Vital agreed with Applicants' representative that the two groups of claims have a substantial number of overlapping features and would not present a serious burden on the Examiner. Mr. Vital stated that he would have the Examiner contact Applicants' representative. The Examiner contacted Applicants' representative on August 4, 2009, to indicate that her Supervisory Patent Examiner agreed with the Examiner that the restriction is proper and that the status of the Final Office Action will not change.

### **II. Election/Restriction**

The Office presents a restriction requirement that states that Applicants must elect a group of claims from the groups designated as follows:

- Group I:       claims 1-5 drawn to a method of synchronizing data, classified in class 707, subclass 203;
- Group II:       claims 24-31 drawn to a middle tier computer used to form a new master data which does not synchronize the master data, classified in class 707, subclass 102.

The Final Office Action states that Group I, claims 1-5 is elected by original presentation. The Office then withdraws claims 24-31 from consideration as being directed to a non-elected invention.

Applicants respectfully traverse and request reconsideration of the restriction requirement under 37 C.F.R. § 1.143.

Applicants respectfully submit that for a restriction to be valid, it must satisfy a two prong test as set forth in MPEP § 803:

There are two criteria for a proper requirement for restriction between patentably distinct inventions:

(A) The inventions must be independent (see MPEP § 802.01, § 806.04, § 808.01) or distinct as claimed (see MPEP § 806.05 - § 806.05(i)); and

(B) There must be a serious burden on the examiner if restriction is required (see MPEP § 803.02, § 806.04(a) - § 806.04(i), § 808.01(a), and § 808.02).

If the search and examination of an entire application can be made without serious burden, the examiner must examine it on the merits, even though it includes claims to independent or distinct inventions.

(MPEP § 803)

In the present case, the restriction requirement does not sufficiently address either criterion. The Final Office Action does not establish that the inventions are distinct other than to baldly state that they do not have to be used together. The Final Office Action also fails to show any serious burden on the Examiner to examine all of the **thirteen** claims in the application. Therefore, the Final Office Action does not set forth a proper requirement for restriction.

To the contrary, the restriction requirement is based on the alleged classification of claims in different classes and subclasses within the U.S. Patent and Trademark Office. First, the Final Office Action classifies the invention of claims 1-5 in class 707, subclass 203. Class 707 is to “DATA PROCESSING: DATABASE AND FILE MANAGEMENT OR DATA STRUCTURES. Subclass 200 is to “FILE OR DATABASE MAINTENANCE.” Subclass 201, which is under subclass 200, is to “Coherency (e.g., same view to multiple users),” and is defined as follows:

This subclass is indented under subclass 200. Subject matter further comprising means or steps for distributed and temporal database management to ensure presentation of the same data or view to one or a plurality of users.

(1) Note. Coherency is related to the maintaining of multiple copies of information in a database or a file in a manner which ensures data integrity amongst the plurality of copies, regardless of whether a single user or plurality of simultaneous users are accessing the information. When a plurality of users is accessing the multiple copies of information, then both

concurrency and coherency concepts may be involved, and the subject matter is classified in the concurrency subclass above.

(2) Note. This subclass is directed to file and database coherency and may include file caching. Caching, per se, however, is classified elsewhere. In addition, cache memory entry replacement strategies are classified elsewhere.

(3) Note. This subclass is directed to file and database coherency and may include management of transactions against a database by means of commit procedures. Transaction management, per se, is classified elsewhere.

(4) Note. This subclass is directed to file and database coherency and may involve access control. Access control in combination with other data processing system methods or apparatus (e.g., memory), computer security, per se, and access control, per se, are classified elsewhere.

(5) Note. This subclass is directed to file and database coherency and may include recitations to shared memory. Managing shared memory, however, is classified elsewhere. Further, data transferring between computers or digital data processing systems is classified elsewhere.

Subclass 203, which is under subclass 201, is to “Version management,” and is defined as follows:

This subclass is indented under subclass 201. Subject matter further comprising means or steps for maintenance and management of multiple copies of database information or files on a computer.

(1) Note. Database concurrency, file or database coherency, and document version management are classified elsewhere in this class. See the SEE OR SEARCH THIS CLASS, SUBCLASS references below and also the line notes and search notes associated therewith.

(2) Note. Software component managing in a software development tool, software upgrading or updating (e.g., plural version management), and software installation are classified elsewhere. See the SEE OR SEARCH CLASS references below and also the line notes and search notes associated therewith.

Applicants contend that if claims 1-5 are properly classified under class 707, subclass 203, then claims 24-31 are also properly classified under class 707, subclass 203.

Independent claims 24 and 28 recite:

24. An apparatus in a middle-tier computer, comprising:  
a processor; and  
a memory coupled to the processor, wherein the memory comprises instructions which, when executed by the processor, cause the processor to:  
receive, via a first software connector, at least one operation from a thin client computer, wherein the thin client computer **stores a copy of**

**master data from a legacy computer** and executes the at least one operation on the copy of the master data;  
sequentially replay the at least one operation on the master data at the legacy computer via a second software connector;  
determine whether the at least one operation is successful; and  
in response to a determination that the at least one operation is successful, **synchronize the master data** by applying the at least one operation via the second software connector to form new master data at the legacy computer, such that in response to a determination that the at least one operation is not successful, the middle-tier computer does not synchronize the master data. [emphasis added]

28. A computer program product comprising a computer recordable medium having a computer readable program recorded thereon, wherein the computer readable program, when executed on a middle tier computer, causes the middle tier computer to:  
receive, via a first software connector, at least one operation from a thin client computer, wherein the thin client computer **stores a copy of master data from a legacy computer** and executes the at least one operation on the copy of the master data;  
sequentially replay the at least one operation on the master data at the legacy computer via a second software connector;  
determine whether the at least one operation is successful; and  
in response to a determination that the at least one operation is successful, **synchronize the master data** by applying the at least one operation via the second software connector to form new master data at the legacy computer, such that in response to a determination that the at least one operation is not successful, the middle-tier computer does not synchronize the master data. [emphasis added]

Claims 24 and 28 also recite features “for distributed and temporal database management to ensure presentation of the same data or view to one or a plurality of users” according to class 707, subclass 201, and also features “for maintenance and management of multiple copies of database information or files on a computer” according to class 707, subclass 203.

The Final Office Action classifies the invention of claims 24-31 in class 707, subclass 102. Subclass 100, under class 707, is to “DATABASE SCHEMA OR DATA STRUCTURE.” Subclass 102, which is under subclass 100, is to “Generating database or data structure (e.g., via user interface),” and is defined as follows:

This subclass is indented under subclass 100. Subject matter including means or steps for generating database schema and data structures.

(1) Note. This subclass accepts operator interface features for data structure development environments. Operator interfaces, per se, are classified elsewhere.

The Final Office Action provides no explanation or reasoning as to why the features of claims 24-31 are somehow directed to generating database schema and data structures. There are no features in claims 24-31 directed to a user interface or specifically directed to **generating** a database schema. For these reasons, Applicants submit that claims 24-31 should be classified in class 707, subclass 203, just as claims 1-5 are.

Furthermore, the Final Office Action states that the invention of claims 24-31 is drawn to a middle tier computer (the first non-legacy computer of claim 1) used to “form” a new master data (synchronized from the copy of the master data by applying the at least one operation) which does not synchronize the master data. However, claim 24, for example, clearly recites “in response to a determination that the at least one operation is successful, **synchronize the master data** by applying the at least one operation via the second software connector to form new master data at the legacy computer” [emphasis added]. Claims 24-31 recite the substantially similar features as claims 1-5. The first non-legacy computer of claim 1 and the middle-tier computer of claim 24, for example, refer to the same computer in the embodiments in the disclosure. Claims 1-5 and claims 24-31 all recite in response to a determination that the at least one operation is successful, synchronizing the master data by applying the at least one operation via the second software connector to form new master data at the legacy computer. Therefore, Applicants respectfully disagree with the Examiner’s characterization that the subject matter of claims 24-31 “can be used together but they don’t have to be together.”

Applicants discussed the restriction requirement with the Examiner in a telephone interview on July 21, 2009. In the Interview Summary issued July 27, 2009, the Examiner states that claim 24 recites that if the operation is not successful, the middle-tier computer does not synchronize the master data. The Examiner then concludes that this clearly shifts the focus to the middle-tier, which is allegedly classified in class 707, subclass 102 (data structure), instead of the focus on the novelty of claim 1, which is the data synchronization. Applicants respectfully disagree. Claim 24 clearly recites “in response to a determination that the at least one operation is successful, **synchronize the master data** by applying the at least one operation via the second software connector to

form new master data at the legacy computer” [emphasis added]. Therefore, claim 24 cannot possibly be interpreted to not have a focus of synchronization.

Furthermore, the “focus” of claims 24-31 to “data structure” proposed by the Examiner appears to be completely arbitrary, because claim 24 does not include any significant features concerning a new database structure, a structured language document, a flat file format, a multi-dimensional database, a user interface for a data structure development environment, or the like. It appears that the separate classification of claims 24-31 is a straw man intended to support the restriction requirement that otherwise has no merit.

Still further, claim 5, which is in the first group, includes the same feature of responsive to a determination that the at least one operation is not successful, not synchronizing the master data. If the Examiner interprets this feature as shifting the focus of novelty with respect to claim 24, then the Examiner **must** make the same interpretation for claim 5. It appears that the Examiner is picking and choosing features that define the “focus” of the claims while ignoring other features, even though both groups of claims include all of the features in question. For example, the first group (claims 1-5) clearly includes the features of a middle-tier computer (the first non-legacy computer in claim 1) and responsive to a determination that the at least one operation is not successful, not synchronizing the master data (claim 5), while the second group (claims 24-31) clearly includes the feature of synchronizing the master data by applying the at least one operation (independent claims 24 and 28). One cannot select features from the first group to define the “focus” of the invention and ignore those same features present in the second group.

Moreover, the restriction requirement does not show any serious burden on the Examiner to examine all of the claims in the application. For instance, claim 1 recites, “storing a master data in at least one legacy computer system,” “storing a copy of the master data in a second non-legacy computer,” and “executing, by said second non-legacy computer, at least one operation on said copy of the master data,” and claim 24 recites, “wherein the thin client computer stores a copy of master data from a legacy computer and executes the at least one operation on the copy of the master data.” Claim 1 recites, “sending, by said second non-legacy computer, said at least one operation to

said first non-legacy computer,” and claim 24 recites, “receive, via a first software connector, at least one operation from a thin client computer.” Claim 1 recites, “replaying, by said first non-legacy computer, said at least one operation,” “determining whether the at least one operation is successful,” and “in response to a determination that the at least one operation is successful, synchronizing said master data by applying said at least one operation to form a modified copy of the master data,” and claim 24 recites, “sequentially replay the at least one operation on the master data at the legacy computer via a second software connector,” “determine whether the at least one operation is successful,” and “in response to a determination that the at least one operation is successful, synchronize the master data by applying the at least one operation via the second software connector to form new master data at the legacy computer.” One cannot search and examine the above features of claim 1 without also searching and examining the above features of claim 24. The same is true for all of claims 24-31. Thus, there clearly is no burden on the Examiner to examine claims 24-31, and the restriction is improper.

Thus, for the reasons set forth above, Applicants respectfully submit that the Examiner has not shown that the inventions are distinct and has failed to establish a serious burden in examining all of the claims set forth in the application. Therefore, the Examiner has not set forth a proper requirement for restriction as required by the MPEP § 803.

### **III. 35 U.S.C. § 112, Alleged Indefiniteness of Claims 1-5**

The Office rejects claims 1-5 under 35 U.S.C. § 112, second paragraph, as allegedly failing to particularly point out and distinctly claim the subject matter that Applicants regard as the invention. Applicants respectfully traverse this rejection.

With respect to claim 1, the Office Action states that it is unclear to the Examiner how the non-legacy computer executes an operation on a master version of data stored on a legacy computer, since the body of the claim does not clearly show how the legacy and non-legacy computers communicate. However, it is not the role of the claims to teach one skilled in the art to reproduce the invention, but rather to define the legal metes and bounds of the invention. *In re Rainer*, 305 F.2d 505, 509, 134 U.S.P.Q. 343, 346 (C.C.P.A. 1962).

If the metes and bounds of the claimed invention are clearly ascertainable, then the claim cannot be properly rejected as “vague” or “indefinite” under 35 U.S.C. § 112, second paragraph. In this case, the scope of claim 1 is clear, even if the manner of implementation is unclear to the Examiner. Whether the claim leaves unclear the manner in which the feature of a non-legacy computer executing an operation on a master version of data stored on a legacy computer may be implemented is irrelevant where the claim clearly covers all forms of implementation. *In re Warmerdam*, 33 F.3d 1354, 1361, 31 U.S.P.Q.2d 1754, 1759 (Fed. Cir. 1994).

Throughout the prosecution of this application, the Examiner has not addressed this argument. Therefore, Applicants respectfully request withdrawal of the rejection of claims 1-5 under 35 U.S.C. § 112, second paragraph.

#### **IV. 35 U.S.C. § 103, Alleged Obviousness of Claims 1-5**

The Office rejects claims 1-5 under 35 U.S.C. § 103(a) as allegedly being unpatentable over *Gehman et al.* (U.S. Patent No. 7,136,881) in view of *Grimsrud* (U.S. Patent No. 6,546,437). Applicants respectfully traverse this rejection.

*Gehman* appears to teach a method and system for processing directory events. An event master server 40 records modification messages in a queue 40b. Event master server 40 includes event message provider 40a, which sends the modification messages as event messages to event service server 41, which includes replicate data monitor 41a. Event service server 41 stores sequence numbers for event messages in queue 41b. Replicate data monitor 41a sends event messages to event client server, which includes event notifier 42a and directory client register 42b. In this way, changes to a master directory database may be replicated or synchronized to directory clients. That is, changes at a master directory database are perpetuated down to clients.

In contradistinction, the present invention provides a method of synchronizing data in a distributed data processing system. The method stores a master data in at least one legacy computer system and enables a first non-legacy computer to support synchronization. A second non-legacy computer (e.g., a client) stores a copy of the master data in a second non-legacy computer, executes at least one operation on the copy of the master data, and sends the at least one operation to the first non-legacy computer.



The first non-legacy computer executes the at least one operation on the master data at the at least one legacy computer. The method determines whether the first non-legacy computer successfully executed the at least one operation and synchronizes the master data by applying the at least one operation in response to the first non-legacy computer successfully executed the at least one operation.

*Gehman* does not teach or suggest executing at least one operation on a copy of the master data in a second non-legacy computer. Rather, *Gehman* teaches perpetuating changes **from** the master directory database **to** the clients, not the other way around. Therefore, it follows that *Gehman* also fails to teach sending, by the second non-legacy computer, the at least one operation to the first non-legacy computer and executing, by the first non-legacy computer, the at least one operation on the master data at the at least one legacy computer, because *Gehman* teaches an event service server that perpetuates changes from the master directory database to clients and fails to teach a non-legacy computer that receives changes made at the client and executes those changes on a legacy computer.

In response to the above argument, the Office Action states that *Gehman* teaches manipulating data within master directory database and replicating the manipulated data within master directory event system to a client corresponding to the copy of master data to a non-legacy computer at col. 3, lines 40-55. Applicant notes that this portion of *Gehman* reinforces Applicant's argument, because *Gehman* clearly teaches that only the master data is manipulated at the legacy computer, that manipulations made at the legacy computer are replicated to the client computer, and that no manipulations are made on the replicated data at the client computer. Therefore, not only does *Gehman* not teach executing at least one operation on a copy of the master data in a second non-legacy computer, but there is no need to synchronize such manipulations back up to the legacy computer.

Furthermore, *Gehman* does not teach or suggest a determination of whether the first non-legacy computer successfully executed the at least one operation. The Final Office Action alleges that *Gehman* teaches such a determination because *Gehman* teaches an option of "yes" in step S96 in FIG. 3B, which is reproduced as follows:

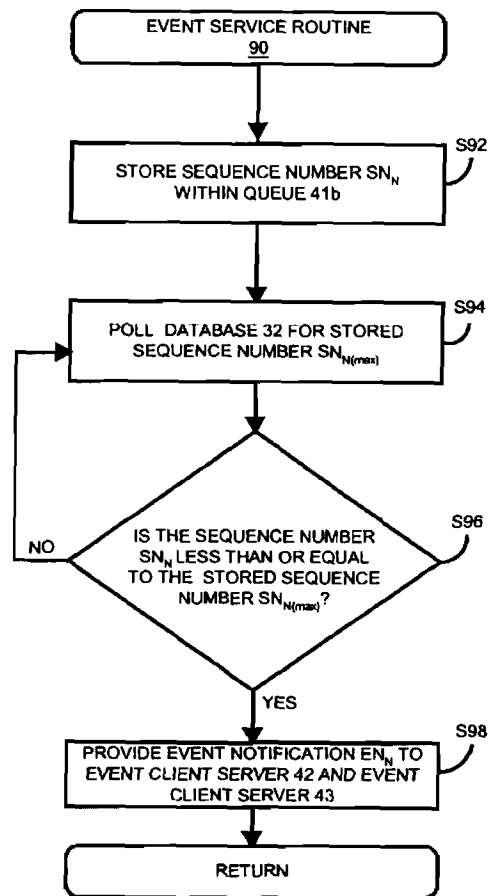


FIG. 3B

As seen in FIG. 3B of *Gehman*, step S96 determines whether a sequence number is less than or equal to the stored sequence number. This determination decides whether to send an event notification to an event client server. *Gehman* states:

Referring additionally to FIG. 3B, during stage S92 of routine 90, replicate data monitor 41a stores sequence number  $SN_N$  within queue 41b. Replicate data monitor 41a proceeds to stage S94 of routine 90 to poll replicate directory database 32 for a sequence number  $SN_{MAX}$  stored therein. Those of ordinary skill in the art will appreciate there can be a significant gap of time from a completion of the writing of sequence number  $SN_N$  to master directory database 31 by event message provider 40a and a subsequent completion of the replication of all of the data within master directory database 31, including the manipulated data and sequence number  $SN_N$ , to replicate directory database 32. As such, those of ordinary skill in the art will further appreciate that sequence number  $SN_N$

being greater than sequence number  $SN_{MAX}$  is an indication that the replication of all of the data within master directory database 31, including the manipulated data and sequence number  $SN_N$ , has not occurred. Conversely, those of ordinary skill in the art will further appreciate that sequence number  $SN_N$  being less than or equal to sequence number  $SN_{MAX}$  is an indication that the replication of all of the data within master directory database 31, including the manipulated data and sequence number  $SN_N$ , has occurred.

Thus, during stage S96 of routine 90, replicate data monitor 41a compares sequence number  $SN_N$  and sequence number  $SN_{MAX}$  (to determine if sequence number  $SN_N$  is less than or equal to sequence number  $SN_{MAX}$ . If sequence number  $SN_N$  is greater than sequence number  $SN_{MAX}$ , replicate data monitor 41a loops back to stage S94. If sequence number  $SN_N$  is less than or equal to sequence number  $SN_{MAX}$ , replicate data monitor 41a proceeds to stage S98 to provide event notification  $EN_N$  (or an edited version thereof) to event client server 42 and event client server 43.

*Gehman*, col. 4, line 39, to col. 5, line 3. Therefore, *Gehman* teaches a determination concerning the sequence of event notifications, but does not teach a determination of whether a first non-legacy computer successfully executes at least one operation on a legacy computer, wherein the at least one operation is received from a second non-legacy computer.

In response to the above argument, the Office Action states that *Gehman* teaches such a determination in a “YES” and “NO” in step S82 of FIG. 3A, which is as follows:

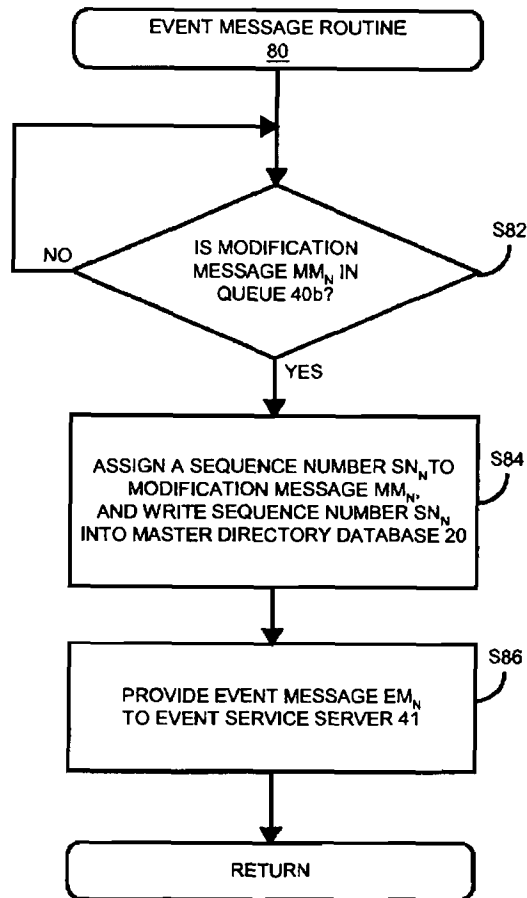
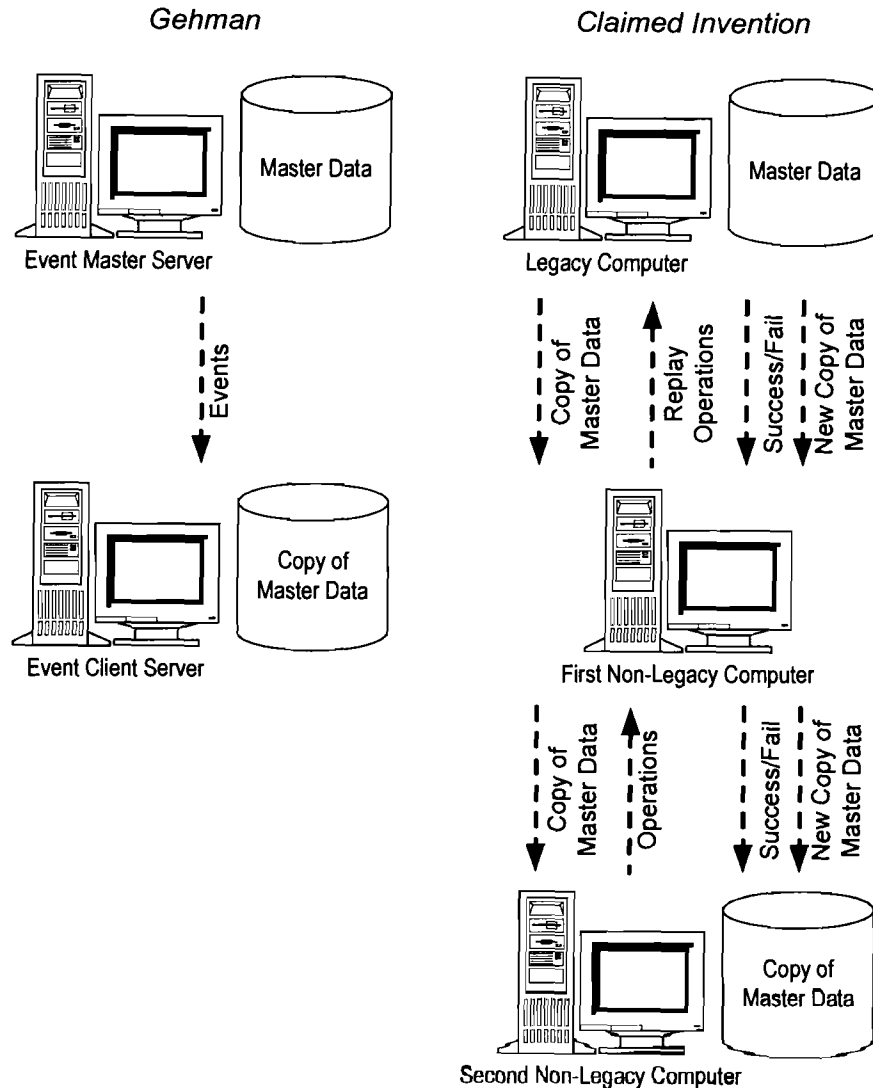


FIG. 3A

Clearly, in step S82, *Gehman* teaches a determination as to whether a modification message exists in a queue to be applied to the master data at the legacy computer. *Gehman* does not teach that such a modification message corresponds to execution of operation on a **copy** of master data at a non-legacy computer. Furthermore, the Office Action proffers no technical analysis or explanation as to why existence of a modification message in a queue is somehow equivalent to a determination of whether at least one operation received from a non-legacy computer that was executed on a copy of master data was successfully replayed on master data at a legacy computer.

As shown in FIG. 2 of *Gehman*, changes at the event master server are perpetuated to the event client server using event messages. The distinctions between *Gehman* and the presently claimed invention are illustrated as follows:



According to claims 1-5, the method of synchronization provides a way for the second non-legacy computer to perform operations on a local copy of the master data and **also** to have those operations replayed by a first non-legacy computer. While *Gehman* may be similar in that changes to master data are perpetuated to other computers, *Gehman* does not teach the **specific combination of features** recited in claims 1-5.

Still further, the Final Office Action acknowledges that *Gehman* does not teach that one computer is a legacy computer and other computers are non-legacy computers. The Final Office Action alleges that *Grimsrud* generally teaches both legacy and non-legacy computers and concludes that it would have been obvious to a person of ordinary skill in the art to incorporate legacy and non-legacy computers in *Gehman*. Applicants respectfully disagree.

*Grimsrud* teaches communication between a computer and a peripheral with a legacy failure control mechanism. *Grimsrud* teaches a computer 174 connected to a peripheral 176 via an advanced technology attachment (ATA) interface 178. The computer 174 may send a request for information to the peripheral, and the peripheral may send a reply to the computer that causes the computer to not use the peripheral. See *Grimsrud*, col. 1, lines 30-56. The computer and peripheral can be any combination of legacy and non-legacy in various embodiments of *Grimsrud*.

However, *Grimsrud* does not teach a person of ordinary skill in the art to modify *Gehman* to include legacy and non-legacy computers. There is no problem in *Gehman* for which *Grimsrud* is a solution. The Final Office Action proposes that a person of ordinary skill in the art would have been motivated to combine the teachings of *Gehman* and *Grimsrud* “in order to the method of detecting any changes or any hardware connected to the system.” While this may be a problem solved in *Grimsrud*, this motivation does not apply to *Gehman* in any significant way. Furthermore, even given this motivation, the teachings of *Grimsrud* would not lead a person of ordinary skill in the art to modify the teachings of *Gehman* in such a way that would result in the presently claimed invention, because neither reference teaches or suggests storing a copy of master data in a second non-legacy computer, executing at least one operation on the copy of the master data, sending the at least one operation to a first non-legacy computer, executing the at least one operation on the master data at the at least one legacy computer, determining whether the first non-legacy computer successfully executed the at least one operation, and synchronizing the master data by applying the at least one operation in response to the first non-legacy computer successfully executed the at least one operation, as recited in claim 1, for example. Rather, a combination of *Gehman* and *Grimsrud* would result in a system for perpetuating changes to master data from an event master server to event client servers where one of the servers may have a peripheral with a legacy failure control mechanism.

The applied references, taken individually or in combination, fail to teach or suggest each and every claim feature. Therefore, *Gehman* and *Grimsrud* do not render claim 1 obvious. Because claims 2-5 depend from claim 1, the same distinctions between

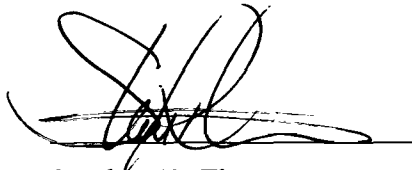
*Gehman* and *Grimsrud* and claim 1 apply for these claims. In addition, claims 2-5 recite additional combinations of features not taught or suggested by the prior art.

Therefore, Applicants respectfully request withdrawal of the rejection of claims 1-5 under 35 U.S.C. § 103(a).

**V. Conclusion**

It is respectfully urged that the subject application is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

Respectfully submitted,

A handwritten signature in black ink, appearing to read 'S. Tkacs', written over a horizontal line.

DATE: August 11, 2009

Stephen R. Tkacs  
Reg. No. 46,430  
Walder Intellectual Property Law, P.C.  
17330 Preston Road, Suite 100B  
Dallas, TX 75252  
(972) 380-9475  
AGENT FOR APPLICANTS